Práctica 2. Factorizaciones LU y Cholesky.

Implementar los métodos de factorización LU y Cholesky para poder resolver sistemas del tipo $Ax = b$.

a) Para una matriz cuadrada A de tamaño $n \times n$ cuyos menores principales sean todos diferentes de cero, deseamos obtener su descomposición $A = LU$, con L matriz triangular inferior y U matriz triangular superior. Lo implementaremos en el **APÉNDICE A**.

La función nos debe devolver, dada A:

- Las matrices L y U correspondientes a la descomposición definida anteriormente.

b) Para una matriz cuadrada A de tamaño $n \times n$ hermítica y definida positiva, deseamos obtener su descomposición $A = BB^*$, con B matriz triangular inferior y diagonal positiva. Lo implementaremos en el **APÉNDICE B**.

La función nos debe devolver, dada A:

- La matriz B correspondiente a la descomposición definida anteriormente.

c) A continuación, descompón la siguiente matriz utilizando ambos métodos y comprueba que el resultado es adecuado.

$$A = \begin{pmatrix} 4 & 1 & 1 \\ 1 & 10 & 0 \\ 1 & 0 & 10 \end{pmatrix}$$

```
A = [4,1,1;1,10,0;1,0,10];
[L,U] = factoLU(A)
```

```
L = 3×3
    1.0000         0         0
    0.2500    1.0000         0
    0.2500   -0.0256    1.0000
U = 3×3
    4.0000    1.0000    1.0000
         0    9.7500   -0.2500
         0         0    9.7436
```

```
[L2,U2] = lu(A)
```

```
L2 = 3×3
    1.0000         0         0
    0.2500    1.0000         0
```

```
    0.2500   -0.0256    1.0000
U2 = 3×3
    4.0000    1.0000    1.0000
         0    9.7500   -0.2500
         0         0    9.7436
```

```
B = factoCholesky(A)
```

```
B = 3×3
    2.0000         0         0
    0.5000    3.1623         0
    0.5000   -0.0791    3.1623
```

```
B2 = chol(A, "lower")
```

```
B2 = 3×3
    2.0000         0         0
    0.5000    3.1225         0
    0.5000   -0.0801    3.1215
```

d) A continuación, factoriza la matriz $A \in GL_{200}(\mathbb{R})$ del archivo 'variables.mat' por el método de Cholesky y resuelve el sistema $Ax = b$.

```
load("variables.mat");
C = factoCholesky(A);
% CC'x=b -> C(C'x)=b -> Cy=b

%Calculamos y
N = size(A,1);
y = (1:N)';
for i = 1:N
    y(i) = (b(i) - C(i,1:i-1)*y(1:i-1)) / C(i,i);
end

%Calculamos x
Ct = C';
x = (1:N)';
for i = N:-1:1
    x(i) = (y(i) - Ct(i,1:i-1)*x(1:i-1)) / Ct(i,i);
end
x
```

```
x = 200×1
   10.6849
    9.8208
    9.2151
    8.7187
    8.2909
    7.9132
    7.5747
    7.2682
    6.9887
    6.7322
      ⋮
```

```
linsolve(A,b)
```

```
ans = 200×1
    1.0000
    1.0000
    1.0000
    1.0000
    1.0000
    1.0000
    1.0000
    1.0000
    1.0000
    1.0000
      ⋮
```

e) Por último, descompón cada una de las siguientes matrices utilizando el método más conveniente de los dos programados en esta práctica. Explica en un comentario por qué utilizas cada método.

$$A = \begin{pmatrix} 7 & 5 & 1 \\ 5 & 8 & 1 \\ 1 & 1 & 3 \end{pmatrix}$$

```
A = [7,5,1;5,8,1;1,1,3];
eig(A)
```

```
ans = 3×1
    2.4605
    2.8095
   12.7300
```

```
tic
factoLU(A);
toc
```

```
Elapsed time is 0.002489 seconds.
```

```
tic
factoCholesky(A);
toc
```

```
Elapsed time is 0.002316 seconds.
```

```
%Elegimos Cholesky porque tarda menos y solo tenemos que calcular y
almacenar una
%matriz en vez de las dos de LU
```

$$A = \begin{pmatrix} 1 & 1 & -3 \\ 1 & 0 & 1 \\ -3 & 1 & -2 \end{pmatrix}$$

```
A = [1,1,-3;1,0,1;-3,1,-2];
eig(A)
```

```
ans = 3×1
```

```
   -4.2965
    0.4020
    2.8945
```

```
% No nos podemos plantear usar Cholesky porque no es definida positiva
```

$$A = \begin{pmatrix} 3 & 2 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 1 & 1 & -1 & 1 \\ 0 & 1 & 8 & 12 \end{pmatrix}$$

```
A = [3,2,2,1;1,2,1,3;1,1,-1,1;0,1,8,12];
issymmetric(A)
```

```
ans = logical
   0
```

```
% No nos podemos plantear usar Cholesky porque no es simétrica
```

Apéndice A. Factorización $A = LU$

```
function [L,U]=factoLU(A)

    [N,M] = size(A);
    L = eye(N,M);
    U = zeros(N,M);
    for i = 1 : N
        for j = 1 : M
            if (i <= j )
                %rellenamos u_ii,...,u_iM
                U(i,j) = A(i,j) - sum(L(i,1:i-1) .* U(1:i-1,j)');
            else
                %rellenamos l_i1,...l_i(i-1)
                L(i,j) = (A(i,j) - sum(L(i,1:j-1) .* U(1:j-1,j)')) / U(j,j);
            end
        end

    end


end
```

Apéndice B. Factorización de Cholesky

```
function B=factoCholesky(A)

N = size(A,1);
B = zeros(N);
```

```
%calculamos elementos de la diagonal
for i = 1:N
    B(i,i) = sqrt(A(i,i) - sum(abs(B(i,1:i-1))).^2);
end

%calculamos elementos por debajo de la diagonal
for i = 1:N
    for j = 1:i-1
        B(i,j) = (A(i,j) - sum(B(i,1:j-1) .* conj(B(j,1:j-1)))) / B(j,j);
    end
end

end
```